Major security component in cars:

* ECU
* CAN Bus
* OBD-II

Microsoft STRIDE:

**S**poofing **T**ampering **R**epudiation **I**nformation disclosure **D**enial of service **E**levation of privilege

Specification for CAN Bus

|  |  |
| --- | --- |
| **Application** | Soft real time system |
| **Control** | Multi-master |
| **Bus Access** | CSMA/CA |
| **Bandwidth** | 500 kbits/s |
| **Data byte per frame** | 0 to 8 |
| **Redundant channel** | Not supported |
| **Physical layer** | Electrical (twisted wires) |
| **Maximum number of nodes** | 30 |

**1. Physical – physical structure (coax, fiber wireless, hubs, repeaters)**

**2. Data link – frames (Ethernet, PPP, switch, bridges)**

3. Network – Packets (IP, ICMP, IPSEC, IGMP)

4. Transport – End- to end connection (TCP, UDP)

5. Session – Synch & send to port (API’s, sockets, winsock)

6. Presentation – syntax layer (SSL, SSH, IMAP, FTP, MPEG, JPEG)

**7. Application – End user level (HTTP, FTP, IRC, SSH ,DNS)**

**\***Highlighted components are the relevant layers for a CAN bus

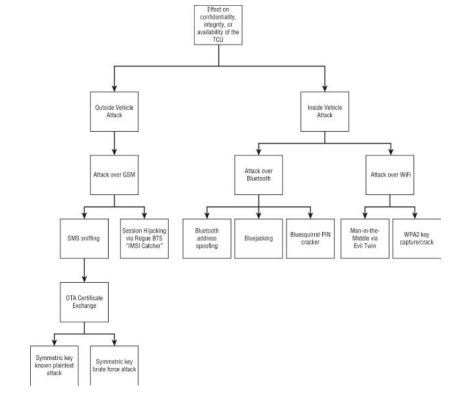
Automobile Bus System in the OSI Model: Example of the CAN bus

* Data link
  + Logical link control (LLC)
    - Acceptance filtering
    - Overload notification
    - Recovery management
  + Medium access control (MAC)
    - Data encapsulation/ decapsulation
    - Frame coding
    - Error detection/ signalling/ handling
  + Physical:
    - Physical signalling (PLS)
      * Bit encoding/ decoding
      * Bit time synchronisation
    - Physical medium attachment (PMA)
      * Driver/ receiver characteristics
    - Medium dependent interface (MDI)
      * Connector

Standard Data Flow Diagram includes:

1. EE: external entity (e.g. Communication endpoints)
2. P: Process (e.g. Function)
3. DF: Data Flow (Transmission of data between two elements)
4. DS: Data Storage (Storage of data, including databases, logs, data output, or files – also includes trust boundaries)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DFD Elements | S | T | R | I | D | E |
| Entity | X |  | x |  |  |  |
| data flow |  | X |  | X | X |  |
| data store |  | X | X | X | X |  |
| process | X | x | X | X | x | X |



|  |  |
| --- | --- |
| STRIDE THREATS | SECURITY ATTRIBUTES |
| Spoofing | Authenticity, Freshness |
| Tampering | Integrity |
| Repudiation | Non- repudiation, Freshness |
| Information Disclosure | Confidentiality, Privacy |
| Denial of Service | Availability |
| Elevation of Privilege | Authorization |

**CAN Bus**

* Used to support communication between Electronic Control Units (ECUs)
* Designed as a closed network, therefore implements no security features such as message encryption or authentication
* Unauthorised party that gains access to the bus can block legitimate messages and transmit illegitimate ones

CAN bus has two major standards to transmit data over a physical medium:

* High speed CAN signalling. ISO 11898-2 (frequent error occurrence)
* Low-speed Fault Tolerant CAN Network. ISO 11898-3 (less error occurrence)

The CAN bus defines the protocols for the physical and data link layers, which enables the communication between the network nodes.

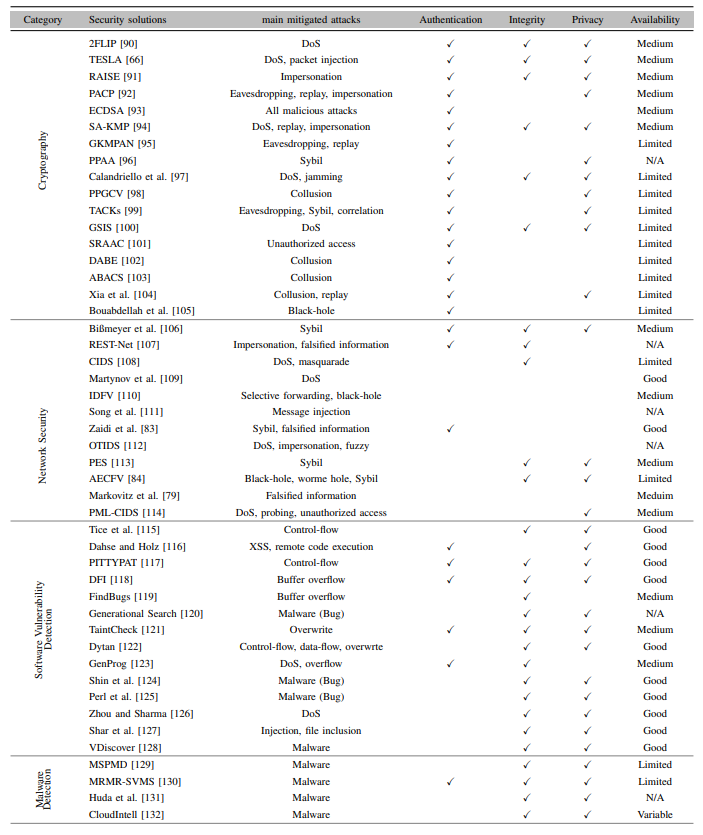
The transport layer is used for Off-Board Communication like Diagnosis and for OBD on these typical automotive area networks. The network and transport layer can be used for Vehicle On-Board Communication in Car2X Communication applications.

Layer 3 routing processes require more infrastructure (e.g. IP stack implementation, software implementation, memory need) and costs investments than layer 2 solutions from an automotive perspective. For in-vehicle control applications which require a very low latency, a layer 2 solution is more pragmatic than a layer 3 solution.

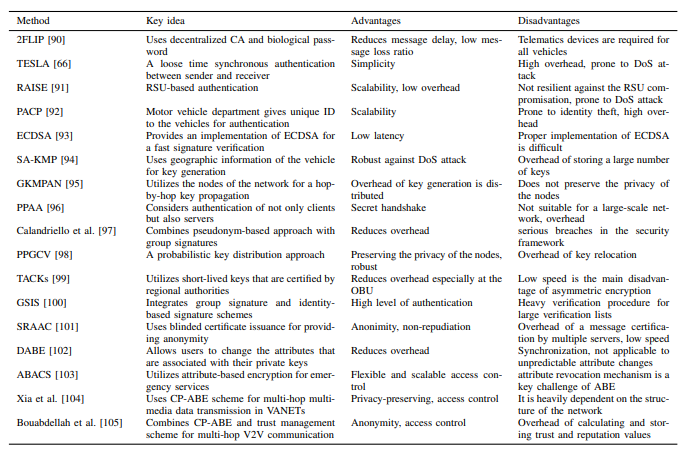
**Common attack types against connected cars:**

* DoS
* DDoS
* Black hole
* Replay
* Sybil
* Impersonation
* Malware
* Falsified information
* Timing

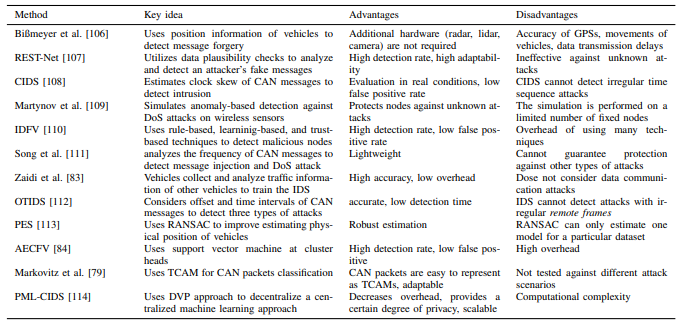
**Existing defence against attacks:**

* Cryptography
  + Symmetric encryption
  + Asymmetric encryption
  + Attribute-based encryption
* Network security
  + Anomaly based detection
  + Signature-based detection
* Software vulnerability detection
  + Dynamic analysis
  + Software testing
  + Machine learning
* Malware detection

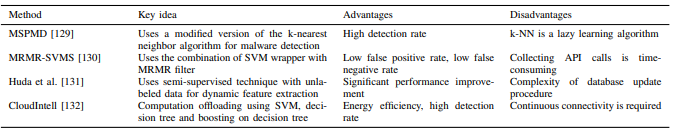
**Key Ideas of Cryptography Defences:**



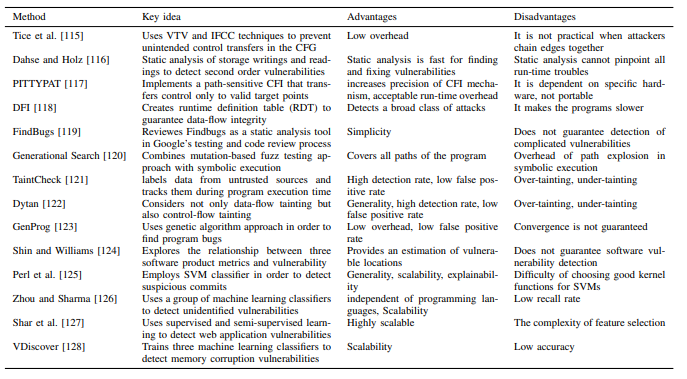
**Key Ideas of Network Security Defences:**



**Key Ideas of Malware Defection Defences:**



**Key Ideas of Software Vulnerability Detection Defences:**



**Mapping of Existing Defence Against Security Attacks:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Cryptography** | **Network Security** | **Software Vulnerability Detection** | **Malware Detection** |
| **DoS** | **X** | **X** | **X** |  |
| **DDoS** | **X** | **X** |  |  |
| **Black-hole** | **X** | **X** |  |  |
| **Replay** | **X** |  |  |  |
| **Sybil** | **X** | **X** |  |  |
| **Impersonation** | **X** | **X** |  |  |
| **Malware** | **X** |  | **X** | **X** |
| **Falsified Information** | **X** | **X** |  |  |
| **Timing** | **X** |  |  |  |

* Synthetic attack generators, using it on the generator
* Main mitigated attack
* Combine the main mitigated attack with STRIDE